

CERTIFICATE OF TRANSMISSION

I hereby certify that this correspondence is being transmitted by fax to the U.S. Patent and Trademark office to fax number 703-746-8327, on February 18, 2004.

David J. McKenzie
Attorney for Applicant

PATENT

Client No. SJ00-00-044

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Kevin F. Smith)	
Serial No.:	09/689,488)	
Filed:	10/12/2000)	
For:	PRESCHEDULING SEQUENTIAL DATA)	Group Art
	PREFETCHES IN A PREEXISTING LRU CACHE)	Unit: 2186
Examiner:	Zhuo H. Li)	

TELEPHONE INTERVIEW AGENDA

Honorable Commissioner of
Patents and Trademarks
Washington, D.C. 20231

Dear Sir:

Applicants would like to hold a telephone interview at 10:30 am EST on Thursday, February 19, 2004, Applicants are submitting this proposed claim amendment. Applicants would appreciate the opportunity to discuss the included claim language in amended claim 1 as compared to allowed claim 20. This amendment includes the novel aspects of the invention from claim 20.

If any issues remain that can be resolved by a telephone conversation, the Examiner is respectfully requested to contact the undersigned.

Respectfully submitted,

David J. McKenzie

David J. McKenzie

Reg. No. 46,919

Attorney for Applicant

Date: February 18, 2004

8 E. Broadway, Suite 600

Salt Lake City, UT 84111

Telephone (801) 994-4646

Fax (801) 531-1929

John 314

*****PROPOSED CLAIM AMENDMENTS – FOR DISCUSSION ONLY*******IN THE CLAIMS**

1. (Currently Amended) A method for scheduling prefetches into a cache of a data storage system, the method comprising:

remotely modeling dynamic operation of the cache in a model; ~~the remotely-modeling~~
including ~~providing~~ a model of data elements currently stored within the cache;

determining whether a data element preceding a requested data element is present in the
cache;

in response to determining whether a data element preceding a requested data element is
present in the cache, assigning a priority value to the requested data element; and

making a cache management decision based upon the model.

20. (Original) A method for scheduling prefetches in a data storage system having a host and a cache, the method comprising the steps of:

providing a cache for caching Input/Output (I/O) data;

providing a prefetch module remote to the cache;

remotely modeling the cache within the prefetch module and determining whether to schedule a prefetch of data into the cache according to the results of the step of remotely modeling the cache, the step of remotely modeling the cache module further comprising:

examining the history of a data element in the cache;

assigning a priority value to the data element according to its history;

comparing that priority value to a predetermined threshold value;

determining a size of memory used in the cache;

periodically fetching an I/O rate of the cache from the cache;

periodically fetching a hit rate of the cache from the cache; and

determining a single reference residency time for a data element within the cache;

intercepting a stream of I/O information from the host to the cache to locate a requested data element;

determining if the requested data element in the stream of I/O information is already present within the cache;

making the requested data element a youngest member of the cache;

determining if the data element preceding the requested data element is present in the cache;

assigning a priority value to the requested data element;

periodically reevaluating the performance of the cache versus an internal model of the cache if the number of I/O requests received by the cache is greater than a predetermined number;

updating the dynamic threshold used in the internal model of the cache if the dynamic threshold value does not adequately model the performance of the cache;

comparing the priority value of the requested data element with the dynamic threshold value;

and

prefetching the requested data element if the priority value of the requested data element is greater than the dynamic threshold value by passing an I/O request of the data element to the cache.